

THIS OPINION WAS NOT WRITTEN FOR PUBLICATION

The opinion in support of the decision being entered today (1) was not written for publication in a law journal and (2) is not binding precedent of the Board.

Paper No.34

UNITED STATES PATENT AND TRADEMARK OFFICE

---

BEFORE THE BOARD OF PATENT APPEALS  
AND INTERFERENCES

---

Ex parte JACK E. OLSON and LINDA C. ELLIOTT

---

Appeal No. 1996-0331  
Application 08/239,942

---

ON BRIEF

---

Before JERRY SMITH, FRAHM and BARRY, Administrative Patent Judges.

FRAHM, Administrative Patent Judge.

DECISION ON APPEAL

Appellants have appealed to the Board from the examiner's final rejection of independent claims 1 and 31, which constitute all of the pending claims before us on appeal. Claims 2 to 30 and 32 to 34 have been canceled.

## BACKGROUND

The subject matter on appeal is directed to the field of database systems, and in particular, to a method of altering an existing data structure in a database system using a change definition language (CDL) (specification, page 2). A CDL is a form of structured query language (SQL), which is a high-level programming language used to access and define data tables, objects, and indexes of the database. Database software typically includes a catalog of definitions for defining the tables and indexes. As indicated in the specification at page 3, it is common that a database system frequently needs to be updated, debugged, or expanded. Appellants have recognized (see specification, page 4) that a problem exists in the prior art that database definitions used by design and development people are different from those used by production-level programmers. To solve this problem, of passing changes from design and development to production and visa versa, appellants provide a CDL which uses a SQL to allow changes to be made to existing definitions at all phases of the development and production cycle (see specification, page 5). In general, appellants' invention recited in independent claims 1 and 31 on appeal seeks to provide a method of altering an existing data structure in a database system which uses a change definition language based on a structured query language. As further discussed, infra, we find that the applied reference to Hawkins fails to teach or suggest at least this

feature (a CDL based on a SQL) as it is recited in the claims on appeal.

Representative claim 1 is reproduced below:

1. A method of altering an existing data structure in a database system,

said existing data structure including a plurality of related tables of columns and rows of data values and further including one or more indexes for said related tables to provide a relational database, said existing data structure being defined by a catalog data structure including tables of tables and indexes, said method comprising the steps of:

a) receiving a change indication signal,

1) said change indication signal encoding one or more statements conforming to a change definition language that is a superset of a SQL structured query language,

2) said change definition language including a plurality of available statements for changing at least one of A) a data specification of a column, B) an owner of a table or an index, and C) a name of a column or an index,

3) said one or more statements specifying one or more requested changes to be made in the existing data structure;

b) testing whether said one or more requested changes can be made in the existing data structure; and

c) if said changes can be made in said existing data structure, then making said changes, or else signaling an error.

The following reference is relied on by the examiner:

John L. Hawkins, (Hawkins), "Clipper Developers Convene," DATA BASED ADVISOR, August 1989, pages 136 et seq. [hereinafter, pages 1 to 6 as referred to by us in the instant opinion].

Claims 1 and 31 stand rejected under 35 U.S.C. § 102(b) as being anticipated by Hawkins.

Rather than repeat the positions of appellants and the examiner, reference is made to the Brief, Reply Brief, and the Answer for the respective details thereof.

### OPINION

In reaching our conclusion on the issues raised in this appeal, we have carefully considered appellants' specification and claims, the applied reference, the respective viewpoints of appellants and the examiner, and all other evidence of record. We find, for the reasons that follow, that the feature recited in claims 1 and 31 on appeal, of providing a method of altering an existing data structure using a change definition language, is not taught by the prior art applied by the examiner. Accordingly, we will reverse the decision of the examiner rejecting claims 1 and 31 on appeal.

At the outset, we note that a prior art reference anticipates a claim only if it discloses, either explicitly or inherently, every limitation of the claimed invention. In re Schreiber, 128 F.3d 1473, 1477, 44 USPQ2d 1429, 1431 (Fed. Cir. 1997). For an unstated element to be found inherent in an anticipating reference, it must exist as a matter of scientific fact and flow naturally from the elements expressly disclosed in the prior art reference. Id. However, inherency may not be established by probabilities or possibilities regarding what may have resulted in the prior art. In re Oelrich, 666 F.2d 578, 581, 212 USPQ 323, 326 (CCPA 1981). Furthermore, "[t]he mere fact that a certain thing *may*

result from a given set of circumstances is not sufficient.” Hansgird v. Kemmer, 102 F.2d 212, 214, 40 USPQ 665, 667 (CCPA 1939) (emphasis in original); see also Ex parte Skinner, 2 USPQ2d 1788, 1789 (BPAI 1986).

Appellants argue (Brief, page 5) that the examiner has not set forth a prima facie case under 35 U.S.C. § 102, because each and every element of the appealed claims is not taught. We agree, and accordingly we cannot sustain the examiner’s rejection of claims 1 and 31 on appeal under 35 U.S.C. § 102(b).

Specifically, appellants argue (Brief, page 4) that Hawkins does not teach altering a data structure, but instead alters just the data itself. We note that representative claim 1 requires "altering an existing data structure," and makes five other references to the data structure (see claim 1 at lines 2, 4, 13, 14 to 15, and 16). Therefore, we find that despite the examiner’s arguments that the preamble should not be given patentable weight (see Answer, pages 3 to 4), the preamble and the body of claim 1 are interwoven to the extent that the preamble depends upon the body of claim 1 for completeness. Thus, we agree with appellants that for Hawkins to anticipate claim 1, Hawkins must explicitly or inherently disclose the recited feature of altering an existing data structure. We find that Hawkins does not teach such a feature.

It is noteworthy that the examiner states that in Hawkins data is changed and a spreadsheet is updated or altered (see Answer, pages 4 and 6), and avoids declaring that Hawkins explicitly teaches altering an existing data structure as claimed. Indeed, our close review of Hawkins reveals that an "Excel spreadsheet and graph" are "updated" and Hayes teaches that "data was changed" (Hayes, page 4), but Hawkins never explicitly discloses that a data structure was altered. The updating of a spreadsheet in Hawkins is the same as changing data, and not altering data structure as required by the claims on appeal. Thus altering a data structure is an unstated step which neither exists as a matter of scientific fact nor flows naturally from the elements expressly disclosed in Hawkins. In this instance, the portions of Hawkins relied upon by the examiner (see page 4, third paragraph; abstract, line 3), merely establish a possibility regarding what may have resulted in the prior art.

Claims 1 and 31 on appeal further require altering data structure by testing whether "requested changes" can be made using a "change definition language" by changing a catalog data structure using a "change statements" (see language of claims 1 and 31 on appeal). Because we find that Hawkins does not anticipate the claims based on a failure to teach changing a data structure, we also find that Hawkins fails to explicitly or inherently disclose the features of making changes in a data structure using change statements, change signals, and a change definition language.

We are not persuaded by the examiner's argument that even though "the verbiage is not identical," Hawkins teaches "receiving a list of change statements" at pages 2 and 4 of the reference

(Answer, page 6) since changes are made to a spreadsheet. This line of reasoning smacks of obviousness, and considerations of obviousness are not applicable to the rejection before us under 35 U.S.C. § 102. We agree with appellants (Brief, page 4; Reply Brief, page 1) that Hawkins discloses changing existing data as opposed to an existing data structure, and that Hawkins does not specifically disclose any use of a "change indication signal," "change

statements," a "list of change statements," or the use of a "change definition language" or a "data description catalog." Thus, the examiner has not shown that every element of the claims is present in Hawkins either explicitly or inherently.

Because an important recited feature of appellant's claims 1 and 31 on appeal, of altering a data structure and not just altering data, is neither expressly nor inherently disclosed by the applied reference to Hawkins, we must conclude that appellants' claims 1 and 31 are not anticipated by Hawkins under the doctrine of inherency. The examiner has failed to make a prima facie case of anticipation.

Lastly, we are not persuaded by the examiner's argument (Answer, page 6) that "Hawkins teaches 'the format of a change description language constituting a superset of an SQL structured query language'" at page 1, paragraph 1. Our close review of Hawkins reveals no quote or teaching of this

Appeal No. 1996-0331  
Application 08/239,942

nature. We are in agreement with appellants (Brief, page 4; Reply Brief, page 1) that Hawkins teaches at page 1 paragraph 1 that Clipper appears to be a superset of dBASE and not a superset of SQL.

Although "Clipper may well be the ultimate superset" (Hawkins, page 1, paragraph 1), the examiner has not adequately shown that Hawkins explicitly or inherently teaches that Clipper is a superset of SQL which reads on appellants' "change definition language."

In view of the foregoing, the decision of the examiner rejecting claims 1 and 31 under 35 U.S.C. § 102(b) is reversed.

REVERSED



Appeal No. 1996-0331  
Application 08/239,942

JERRY SMITH	)	
Administrative Patent Judge	)	
	)	
	)	
	)	BOARD OF PATENT
ERIC FRAHM	)	
Administrative Patent Judge	)	APPEALS AND
	)	
	)	INTERFERENCES
	)	
LANCE LEONARD BARRY	)	
Administrative Patent Judge	)	

EF/pgg  
Arnold, White & Durkee  
D.C. Toedt  
P.O. Box 4433  
Houston, TX 77210